A Cross-Correlation Based Method for Spatial-Temporal Traffic Analysis

Jian Yuan Kevin Mills

Advanced Network Technologies Division
National Institute of Standards and Technology

June 5, 2003 – DRAFT

Abstract

Analyzing spatial-temporal characteristics of traffic in large-scale networks requires both a suitable analysis method and a means to reduce the amount of data that must be collected. Of particular interest would be techniques that reduce the amount of data needed, while simultaneously retaining the ability to monitor spatial-temporal behavior network-wide. In this paper, we propose such a method, motivated by insights about network dynamics at the macroscopic level. We define a weighted vector to build up information about the influence of local behavior over the whole network. By taking advantage of increased correlations arising in large networks, this method might require only a few observation points to capture shifting network-wide patterns over time. This paper explains the principles underlying our proposed method, and describes the associated analytical process.

Keywords: network traffic, timescale, cross-correlation, spatial-temporal pattern, eigenvalue, eigenvector

1 Introduction

Most extant research on network traffic analysis focuses on observing *temporal* dynamics of traffic and effects from user and protocol behavior [1-4]. In such analyses, detailed IP packet traces on individual links reveal the characteristics of network traffic at multiple timescales, e.g., rich scaling dynamics arising over small timescales [3], and self-similarity and long-range dependence at large timescales [4]. Recently, graph wavelets have been proposed for *spatial* traffic analysis with knowledge of aggregate traffic measurements over all links [5]. This method can provide a highly summarized view of traffic load throughout an entire network. Despite these advances, spatial and temporal traffic analysis still presents difficult challenges, not only because large-scale distributed networks exhibit high-dimensional traffic data, but also because current analytical methods require examination of large amounts of data, which can strain memory and computation resources in even the most advanced generation of desktop computers.

Despite these inherent difficulties, investigation of spatial-temporal dynamics in large-scale networks is an important problem because modern society grows increasingly reliant on the Internet, a network of global reach that supports many services and clients. Lacking means to predict, monitor, and adjust spatial-temporal dynamics, Internet Service Providers (ISPs) typically over-provision network capacity, which typically leads

to under-utilized resources on average with overloaded hotspots arising from time to time. Further, the Internet appears increasingly vulnerable to attacks and failures [6, 7]. These factors suggest a crucial requirement to devise and develop promising tools that can monitor network traffic in space and time to identify shifting traffic patterns. Such tools can aid operating and engineering large-scale networks, such as the Internet. While useful network management tools might focus on either offline or online monitoring and analysis, the task of network-wide on-line monitoring presents more stringent requirements for transferring and handling traffic data in a timely fashion.

To support the development of useful network management tools, the networking research community endeavors to devise novel and accurate methods to interpret measurements, and to derive principles for extracting information from raw measurement data. For example, a recent work studies correlations between different network flows in a French scientific network, *Renater* [8]. The study defines a network flow as a packet flow transferred from a given starting router to a given destination router. Many such flows simultaneously transit a large-scale network, leading to underlying *interactions* among the flows. Unfortunately, the effects of such interactions are usually not known, and so cannot contribute to better network engineering and management. The *Renater* study uses methods from random matrix theory (RMT) to analyze cross-correlations between network flows. (RMT methods have been recently used to study correlations in financial data [9].) In essence, RMT compares a random correlation matrix—a correlation matrix constructed from mutually uncorrelated time series—against a correlation matrix for the data under investigation. Deviations between properties of the cross-correlation matrix from the investigation data and the correlations in the random data convey information about "genuine" correlations. In the case of the *Renater* study, the most remarkable deviations arise about the largest eigenvalue and its corresponding eigenvector. The largest eigenvalue is approximately a hundred times larger than the maximum eigenvalue predicted for uncorrelated time series. The largest eigenvalue appears to be associated with a strong correlation over the whole network. In addition, the eigenvector component distribution of the largest eigenvalue deviates significantly from the Gaussian distribution predicted by RMT. Further, the *Renater* study reveals that all components of the eigenvector corresponding to the largest eigenvalue are positive, which implies their collective contribution to the strong correlation. Since all network flows contribute to the eigenvector, the eigenvector can be viewed as the signature of a collective behavior for which all flows are correlated. Thus, the eigenvector might provide an important clue about macroscopic behavior of the underlying interactions. In other words, the predominant information about network dynamics at the macroscopic level can be obtained from the largest eigenvalue and its corresponding eigenvector. This insight might prove very helpful for analyzing spatial-temporal traffic patterns in large-scale networks.

In this paper, we propose a method for spatial-temporal traffic analysis using the eigenvector corresponding to the largest eigenvalue. As the macroscopic pattern emerges from all adaptive behaviors of flows in various directions, hotspots should be exposed, through their correlation information, as the joining points of significantly correlated flows. Note that the details of the components of the eigenvector of the largest eigenvalue reveal this information, with the larger components corresponding to the more correlated flows. Thus, our primary insight is to group eigenvector components corresponding to a destination routing domain (or autonomous system) together to build up information about the influence of the routing domain over the whole network. We define a weight vector for this purpose. Contrasting weights against each other in the weight vector, we not only can summarize a network-wide view of traffic load, but also locate hot spots, and even observe how spatial traffic patterns change from one time period to the next.

While our approach builds upon the *Renater* study, we must solve some special problems related to scale. The *Renater* study assumes complete information from all network connection points, which proves feasible because the *Renater* network contains only about 30 interconnected routers. Arranging for complete coverage of observations in larger networks raises issues of scale, both in gathering data from numerous measurement points and in consuming computation time and memory when analyzing data. In particular, some heavily utilized routers may fail to collect and transfer measurement data. Usually, it is impossible to monitor areas of interest without corresponding measurements from those areas. We exploit correlation increases, arising from *collective response* of the entire network to changes in traffic, to extend our ability to monitor network-wide behavior. This effect has already been observed in the framework of stock correlations, where cross-correlations become more pronounced during volatile periods as compared to calm

periods [9]. Indeed, higher values of the largest eigenvalue occur during periods of high market volatility, which suggests strong collective behavior accompanies high volatility. This connection should have value in our analysis because Internet traffic behavior appears to be nonstationary [10]. An increase in cross-correlation allows us to infer a shift in the spatial-temporal traffic pattern of large areas of interest outside those few areas where measurements are made. This approach could significantly reduce requirements for data, perhaps to the point where monitoring may be performed in real time.

In this paper, we use simulation results to show how our proposed technique might work in a real large-scale network. Our results derive from a simulation model we developed recently to study space-time characteristics of congestion in large networks, and to analyze system behavior as a coherent whole [11]. While capturing essential time details of individual packets and connections, the model accommodates spatial correlations arising from interactions among adaptive transport connections and from variations in user demands. In particular, our model offers a clear-cut framework to analyze spatial-temporal traffic patterns, e.g., where will hotspots develop and how long will they persist? Coupling our new measurement and analysis technique with our existing simulation model allows us to compare weight vectors at different timescales. Using this approach, we can identify the macroscopic pattern at timescales of interest, allowing us to observe that network-wide hotspots become more prominent as increased correlation emerges. First, we try our method assuming complete measurement data, and then we further try our method with only a few observation points. The rest of this paper is structured as three sections. Section II describes our adaptation of the RMT cross-correlation method. In Section III, we delineate our simulation model and show experiment results. We present concluding remarks in Section IV.

2 The Cross-correlation Based Method

Here, we describe how we represent network flow data and how we apply cross-correlation analysis to the data. Then, we describe how we apply RMT (random matrix theory) to investigate cross-correlation throughout a network.

2.1 Representing network flow data

We need to represent packets flowing between distinct source-destination pairs at each sampling interval in our model. Let $\boldsymbol{x} = (\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_N})^T$ denote the flow vector of corresponding packet counts among all N routing domains, observed in starting domains during a given time interval, T, in a large network. Here ^{T} indicates transpose. Each element of this flow vector is itself a vector defining the number of packets flowing into the corresponding domain from each of the other (starting) domains in the network. The method to obtain all flow variables in this vector is to first enumerate all the destination domains and then the starting domains by 1 to N, and group these indices by routing domain: the domains sending to the first domain in the first block, $\boldsymbol{x_1}$, and those sending to the second domain in the second block, $\boldsymbol{x_2}$, and so forth. Thus, we form $\boldsymbol{x}$ with blocks in the order $\boldsymbol{x_1} = (x_{21}, x_{31}, \ldots, x_{N1})^T$, $\boldsymbol{x_2} = (x_{12}, x_{32}, \ldots, x_{N2})^T$, $\boldsymbol{x_3} = (x_{13}, x_{23}, x_{43}, \ldots, x_{N3})^T, \ldots, \boldsymbol{x_N} = (x_{1N}, x_{2N}, \ldots, x_{(N-1)N})^T$, where x_{ij} $(i \neq j)$ represents packet flow from the ith domain to the jth domain. Each flow variable x_{ij} is normalized as f_{ij} by its mean m_{ij} and standard deviation σ_{ij},

$$f_{ij} = (x_{ij} - m_{ij}) / \sigma_{ij}. \quad (1)$$

Then, the normalized flow vector $\boldsymbol{f}$, corresponding to $\boldsymbol{x}$, comprises N normalized subvectors, $\boldsymbol{f_k}$ (k =1, 2, …, N), where each subvector is formed from normalized flow variables f_{ik} ($i \neq k$ and $i \leq N$). If M is the number of observed samples over the observation period of $M \times T$, then $\boldsymbol{f}$ is a $N(N-1) \times M$ matrix.

2.2 Cross-correlation analysis

Cross-correlation analysis is a tool commonly used to analyze multiple time series. We can compute the equal-time cross-correlation matrix $\boldsymbol{C}$ with elements

$$C_{(ij)(kl)} = \left\langle f_{ij}(t) f_{kl}(t) \right\rangle, \quad (2)$$

which measures the correlation between f_{ij} and f_{kl}, where $\langle \cdots \rangle$ denotes a time average over the period studied. The cross-correlation matrix is real and symmetric, with each element falling between –1 and 1. Positive values indicate positive correlation, while negative values indicate an inverse correlation. A zero value denotes lack of correlation.

We can further analyze the correlation matrix $\boldsymbol{C}$ through eigenanalysis [12]. The equation

$$\boldsymbol{C}\boldsymbol{w} = \lambda \boldsymbol{w} \quad (3)$$

defines eigenvalues and eigenvectors, where λ is a scalar, called the eigenvalue. If $\boldsymbol{C}$ is a square K-by-K matrix, e.g., $K = N(N-1)$ in the case of complete coverage, then $\boldsymbol{w}$ is the eigenvector, a nonzero K by 1 vector (a column vector). Eigenvalues and eigenvectors always come in pairs that correspond to each other. This eigenvalue problem has K real eigenvalues, some of which may repeat. An eigenvector is a special kind of vector for the matrix it is associated with, because the action of the underlying operator represented by the matrix takes a particularly simple form on the eigenvector input: namely, simple rescaling by a real number multiple. The eigenvector $\boldsymbol{w^1}$ corresponding to the largest eigenvalue λ_1 often has special significance for many applications. There are various algorithms for the computation of eigenvalues and eigenvectors [12]. Here, we exploit the MATLAB 'eig' command, which uses the QR algorithm to obtain solutions [13].

2.3 Applying random matrix theory

Much of the traffic flowing through the Internet must traverse multiple routing domains. Adaptive behaviors of flows in different directions play a crucial role in forming macroscopic patterns, mostly in a self-organized manner. The cross-correlation matrix contains within itself information about underlying interactions among various flows. In a study of cross-correlations in stock price changes, influence strength is defined as the sum of the cross-correlation coefficients associated with one company in order to compare the role of that company's stock price changes in affecting the entire stock market [14]. Similarly, we can measure the congestion level of the jth domain by summing all cross-correlation coefficients (ignoring autocorrelation) associated with the jth block, i.e., $\sum_i \sum_{k,l} C_{(ij)(kl)}$, $(i \neq j, k \neq l)$. Using this approach in our simulations yielded similar findings as when this technique was applied to study stock markets [9] and the *Renater* network [8]. That is, the majority of the properties of the correlation matrix $\boldsymbol{C}$ conformed to the results predicted by RMT[1]; thus, the correlation coefficients included substantial noise mixed with the information about macroscopic patterns. We found this to hold even when observing network traffic flows in all nodes, and to hold more strongly in cases where we observed network traffic in only a sparse number of nodes. From this, we conclude that we are more likely to find less noise (and more information) in cases that deviate from the RMT predictions. Such cases can be found by filtering the information about structural correlations through eigenanalysis.

The components of the eigenvector $\boldsymbol{w^1}$ of the largest eigenvalue λ_1 represent the corresponding flows' influences on macroscopic behavior, abstracted from the matrix $\boldsymbol{C}$ into the pair (λ_1, $\boldsymbol{w^1}$). The eigenvector $\boldsymbol{w^1}$ comprises N subvectors, i.e., $\boldsymbol{w^1} = (\boldsymbol{w^1_1}, \boldsymbol{w^1_2}, \ldots, \boldsymbol{w^1_N})^T$. The kth subvector, corresponding to the kth domain, is formed from components w^1_{ik} ($i \neq k$ and $i \leq N$) representing the ith domain's contribution to the kth domain. We consider the square of each component, $(w^1_{ik})^2$, instead of w^1_{ik} itself because $\sum_{i,k} (w^1_{ik})^2 = 1$ [15]. We define the weight S_k ($k = 1, 2, \ldots, N$) to be the sum of all $(w^1_{ik})^2$ in the kth subvector $\boldsymbol{w^1_k}$.

$$S_k = \sum_{i(\neq k)}^{N} (w^1_{ik})^2 . \quad (4)$$

[1] In the *Renater* study, the eigenvalues' distribution and their spacing distribution follow approximately the predictions of RMT. And, the eigenvectors corresponding to most eigenvalues are in agreement with the results of RMT.

In the case of complete observations in all routing domains, S_k represents the relative strength of the contributions of the flows towards the kth routing domain. Thus, the knowledge of weight vector $\boldsymbol{S} = (S_1, S_2, \ldots, S_N)$ across varying k constitutes one summary view of network-wide traffic load.

When analyzing the spatial-temporal traffic pattern of a large-scale network, the cross-correlation matrix $\boldsymbol{C}$ can be a very large object. Usually, floating-point operations on the order of K^3 are required to find eigenvalues and eigenvectors [12]. Thus, even if the analysis yields information results, it appears impractical to monitor the spatial and temporal pattern of large-scale networks using this method under complete coverage of observations. We exploit the property of the increased correlation in order to reduce data requirements, filling the flow vector $\boldsymbol{x}$ just with traffic measured in a few domains. This insight might allow us to infer the traffic pattern shift in real time for large areas of interest from observations in a few distant locations.

3 Experimental Analysis

In this section, we show some experimental results after a brief description of our simulation model. We first discuss the timescale of interest. Assuming complete coverage of observations, we also discuss qualitatively the increased correlation at the timescale of interest. Then we demonstrate our method for cases of sparse observation points within a larger network structure.

3.1 Simulation model

Network simulation plays a key role in building an understanding of network behavior. Choosing a proper level of abstraction from a model depends very much on the objective. Studying large-scale characteristics and collective phenomena seems to require simulating networks at large scale. Appropriate models for these purposes should also include substantial detail representing protocol mechanisms across several layers of functionality, yet must be restricted in space and time in order to be computationally tractable. We propose a modeling approach that maintains the individual identity of packets to produce the full-duplex "ripple effect" at the packet level, and that can also accommodate spatial correlations in a regular network structure [11, 16]. Our model has been tested successfully against current understanding of the timescale dynamics of network traffic, and has been used to show a significant influence of spatial span on correlation structure.

The topology of our model comprises numbers of interconnected domains, as shown in Figure 1. Each domain has two tiers: an upper tier for routers and a lower tier for hosts. Each router is attached to an equal number of sources (100 in this paper), and to a variable number of hosts ($\leq$ 500 in this paper) acting as receivers. Our model operates at the packet level. In this paper, each source models traffic generation as an ON/OFF process, which alternates between wake and sleep periods with average durations λ_{on} and λ_{off}, and with the same shape parameter α of the Pareto distribution [11] for both ON and OFF processes. When a source initiates a connection (ON period), a destination routing domain (differing from the source domain) is chosen randomly and uniformly. To store and forward packets, which travel a constant, shortest path between a source-destination pair for each flow, routers maintain a queue of limited length (160 packets/router here), where arriving packets are stored until they can be processed: first-in, first-out. For convenience, in this paper we assume that every discrete simulation time-step is 1 millisecond. If a source is in the ON period, the source can create one packet every millisecond under the control of TCP, and forward it to the buffer of its directly attached router. However, each router can forward multiple packets (10 here) during one millisecond. This simulates the difference between access links and backbone links in a hierarchically structured network.

With our model, we can simulate spatial and temporal traffic dynamics through high user variability (α = 1.5, λ_{on} = 50 and λ_{off} = 3000), and through adaptive transport connections. We assume a network with N = 25 domains (Figure 1). Note that there is no structural bottleneck in our model because routing assumes a periodic boundary condition, which allows the edges of our grid topology to form a closed structure [11].

Given homogeneous variation of traffic demand in space and time, heavily congested subnets are induced only infrequently. To deliberately induce congestion, we let one selected domain have an additional two percent probability for selection as the destination domain. This is a natural way to change the network-wide traffic demand at longer timescale. We measure the most congested domain, which has the maximum number of connections destined for itself. Figure 2 shows that the address, y_A (= 1, 2, …, 25), of the most congested domain changes over time. During the first period, the 19th domain is the most congested one in the network. The 7th domain is selected as a new location to induce the next congestion at t = 800s, but the second period of congestion actually starts from t = 1232.9s. The 19th domain is selected again as the hotspot at t = 1600s, but the third period of congestion starts 542.2s into the second period. Obviously, this congestion-induction technique offers an easily interpreted framework to analyze spatial-temporal pattern shifts driven by varying traffic demand.

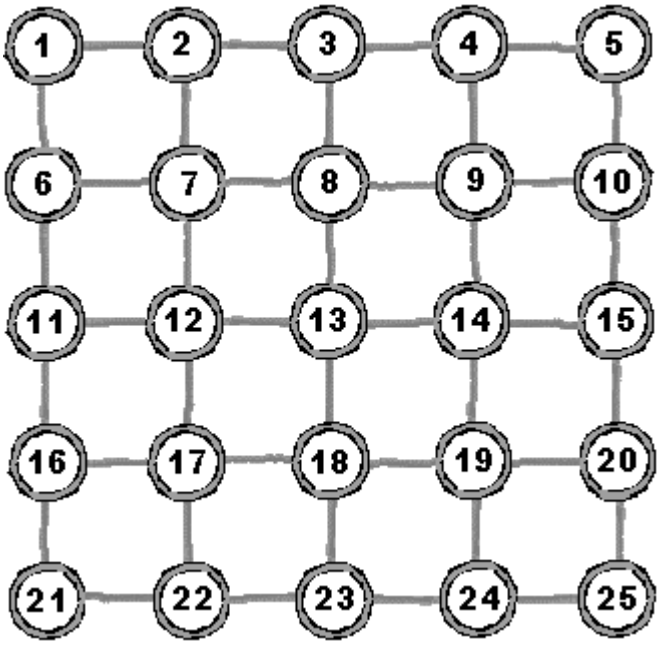

Figure 1: The network structure with 25 routing domains

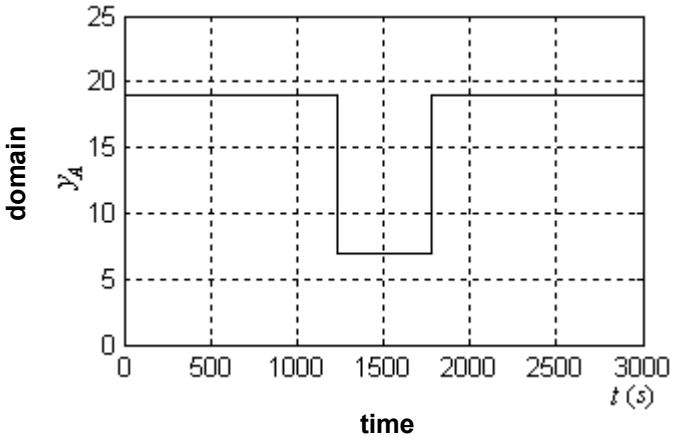

Figure 2: The most congested domain changes over time

3.2 Timescale of interest

In the above simulation, we observe at granularity of 100ms (i.e., every 100 model time steps) every fine-grain flow between all domain pairs, filling the flow vector $\boldsymbol{x}$ with 600 variables (24 destination domains for each of the 25 source domains). Such complete coverage of observation allows us to analyze cross-correlations of all flows aggregated at various time granularities T.

When focusing on network-wide behavior, the timescale of interest should not be fine-grained. The microscopic fluctuations observed at shorter timescales usually reflect local details, while the driving force of traffic demand seems to vary over much longer timescales. The timescale of interest in our experiments appears at a middle range, similar to the concept of a critical timescale beyond which the traffic fluctuation is supposed to exhibit greater influence [17]. At this middle timescale, macroscopic behavior forms a connecting link between microscopic fluctuations and the longer-range driving force of variations in traffic demand. This expected coherence emerges as a result of adaptive behaviors of flows in different directions, but continues to shift its spatial-temporal pattern under the force of traffic demand.

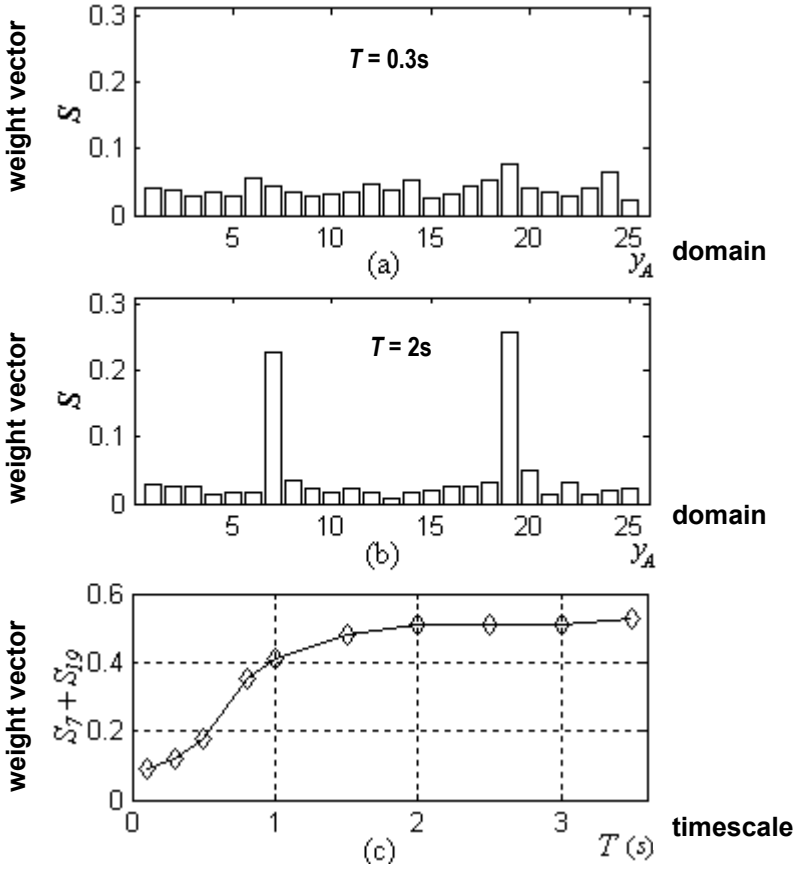

Figure 3: Two weight vectors at T = 0.3s (a) and T = 2s (b), and (c) the sum of S_{19} and S_7 changing at different timescales

We first calculate the weight vector $\boldsymbol{S}$ with M data points (M = 200 in this paper), which span a first period ($M/2$ points) and a second period ($M/2$ points). Two weight vectors are calculated at the aggregated levels T = 0.3s and T = 2s, and shown respectively in Figure 3(a) and 3(b). The weight vector

with T = 2s shows two prominent weights at the 19th and 7th domains (S_{19} and S_7), revealing the network-wide pattern of congestion arising in these two domains. However, the pattern does not appear when T = 0.3s. To clarify the role of timescale here, we further show the sum of S_{19} and S_7 at different aggregated levels in Figure 3(c). We find that the sum of S_{19} and S_7 gradually increases as T increases, but becomes saturated from T = 2s. To show how the spatial traffic pattern changes, we calculate the weight vector $\boldsymbol{S}$ using M data points within a moving time window MT from one time period to the next. Figure 4 shows the weight vector $\boldsymbol{S}$ evolving with T = 2s and with the time window MT (= 200 × 2s = 400s) sliding ahead every 40s. The time axis indicates the end of the moving time window. This technique provides a useful way to observe network-wide congestion patterns shifting over time.

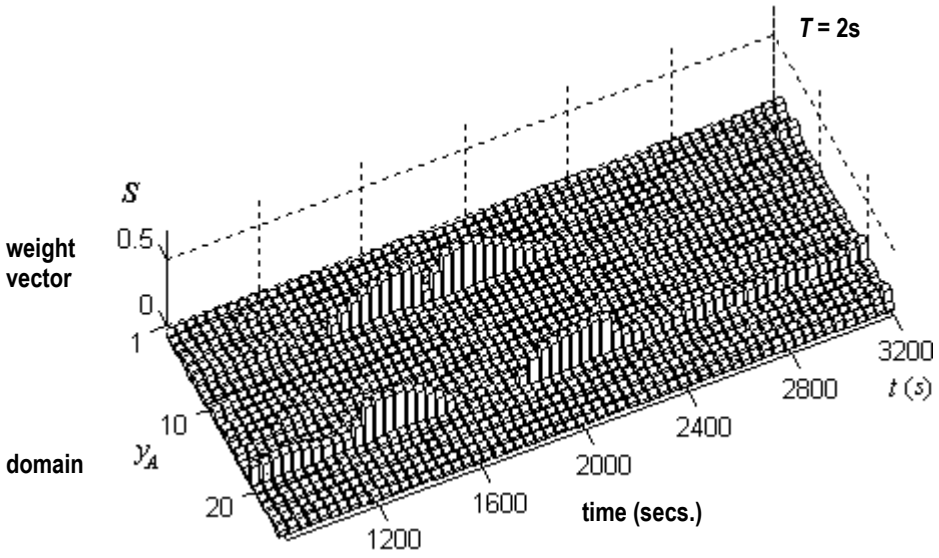

Figure 4: The spatial-temporal pattern evolving with T = 2s

3.3 Increased correlation

Figure 5(a) shows the S_{19} (dotted line), S_7 (dashed line), and the sum of S_{19} and S_7 (solid line), which are calculated with T = 1.5s and with the time window MT (= 200 × 1.5s = 300s) sliding ahead every 30s. And the corresponding λ_1 shows in Figure 5(b). While S_{19} and S_7 are distinguishable in three periods, both become enhanced during periods of pattern shifting. The sum of S_{19} and S_7, and the largest eigenvalue λ_1 undulate in the same way, and reach higher values during periods of pattern shifting than during calm periods. The increased correlation in the simulation data results from a *collective response* of the entire network to changes in traffic demand. During transient periods, flows in different directions have to adapt their behaviors to the changing impulse of the driving force, and continue to react to each other until they reach collectively a new coherent pattern. With the measurement and analysis method, as outlined above, applied at the appropriate timescale, as cross-correlations become more pronounced, traffic patterns over the whole system become more visible.

One might hypothesize that system-wide visibility depends on choosing an appropriate timescale. For example, observe the system at a coarser timescale of T = 3s, as shown in Figure 6. We show S_{19} (dotted line), S_7 (dashed line), and the sum of S_{19} and S_7 (solid line) in Figure 6(a), and the corresponding λ_1 in Figure 6(b). Comparing Figures 5 and 6, we find that two transient processes seem to converge gradually as T increases, and the second period becomes indistinct as if a hotspot appears on the 7th domain for some time. When T is above 4s (not shown), congestion on the 19th domain never appears to diminish.

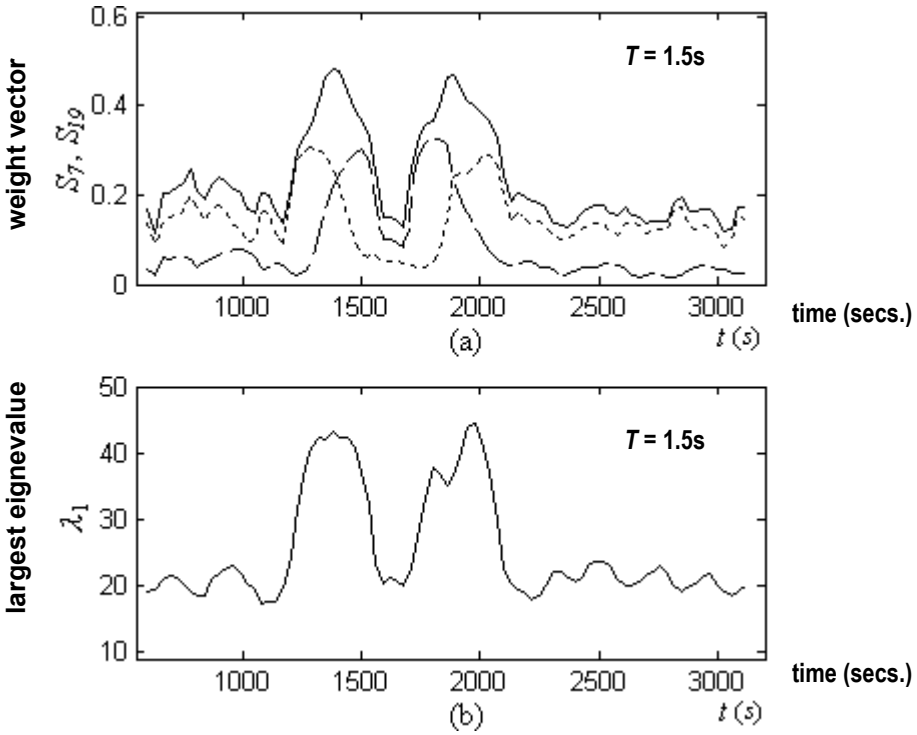

Figure 5: (a) S_{19} (dotted line), S_7 (dashed line), and the sum of S_{19} and S_7 (solid line), and (b) the largest eigenvalue λ_1 with T = 1.5s

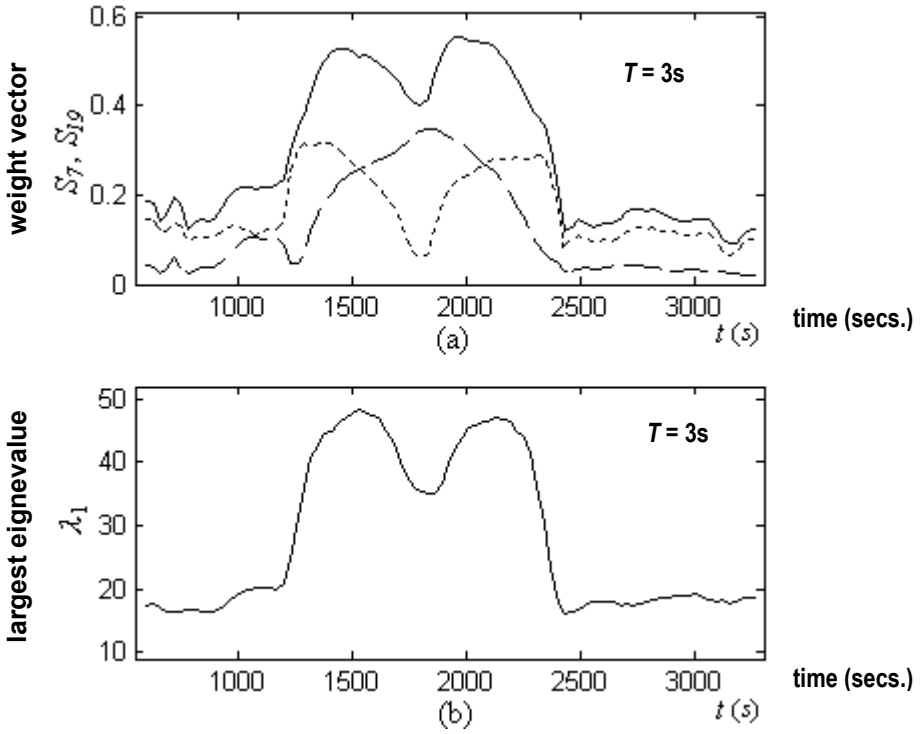

Figure 6: (a) S_{19} (dotted line), S_7 (dashed line), and the sum of S_{19} and S_7 (solid line), and (b) the largest eigenvalue λ_1 with T = 3s

3.4 Sparse observation posts

While our proposed data analysis method provides substantial visibility into network-wide behavior at the critical timescale, it is impractical to collect fine-grain traces for every source-destination pair in a large network. Even if complete observations can be arranged, challenges remain, such as obtaining reliable data transfer to the analysis point and implementing processing power sufficient to analyze the data within a meaningful time. In particular, some heavily utilized routers may fail to collect and transfer data, but often happen to be the parts of interest to monitor (due to their congested nature). Given these real constraints, it would be appealing to reduce the amount of data to transfer and process, while retaining our ability to monitor the network-wide behavior.

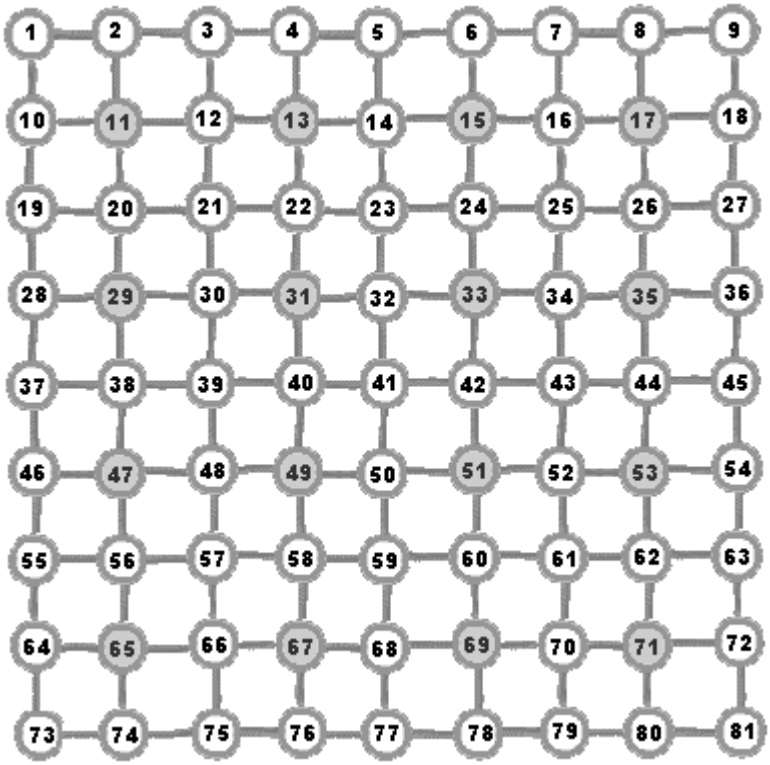

Figure 7: The larger network with 81 domains and separated observation points

It might prove easy to design sample-based techniques suitable to identify network-wide patterns that remain invariant for a long time. However, when traffic demands vary over a large dynamic space-time range, these same techniques might fail to detect more quickly changing patterns. By taking advantage of increased correlation arising in volatile periods, we might be able to use a sample-based version of our proposed method to identify shifting network-wide congestion patterns. In the following, we provide some preliminary results regarding this idea.

Figure 7 shows a larger network with 81 domains and L (= 16) observation points (shaded). For each source we use the following traffic-generation parameters: α = 1.5, λ_{on} = 50 and λ_{off} = 5000. We record traffic flowing out from each observation point to all other domains with T = 2.1s, and we fill the flow vector $\boldsymbol{x}$ with $L \times (N-1)$ (= 16 × 80 = 1280) variables, representing a substantial reduction from the 6480 variables needed for complete monitoring. We select a total of four domains as hotspots, and increase congestion in two of the domains in each of two different time periods. Figure 8(a) shows how the most congested

domains, y_A (= 1, 2, …, 81), change over time. In the first period (up to about 1830s), we arrange for the 21st and 61st domains to be most congested. In the second period (after 1830s), we arrange for the 25th and 57th domains to be most congested. We then calculate the weight vector $\boldsymbol{S}$ with 200 data points spanning the two periods. In Figure 8(b), the weight vector shows four prominent weights at the 21st, 25th, 57th and 61st domains (S_{21}, S_{25}, S_{57} and S_{61}), and thus reveals the network-wide pattern that we stimulated. From this, we infer that such patterns can be detected even without complete observations. Note also, that this technique managed to find the congestion pattern without sampling packets flowing out of the congested domains.

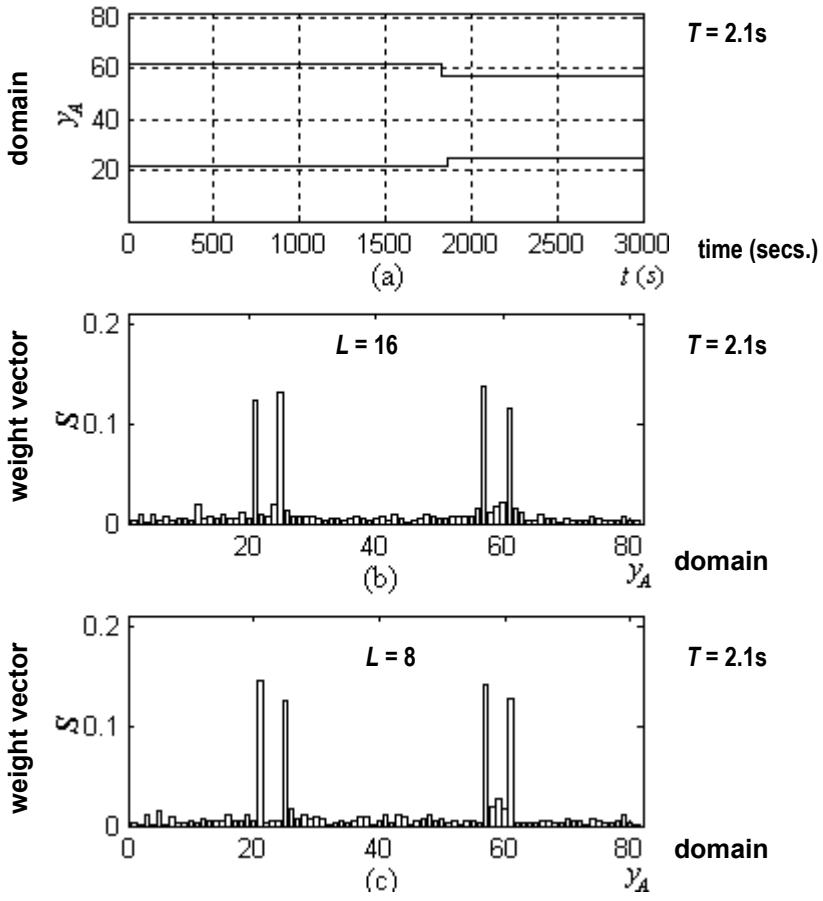

Figure 8: (a) The most congested domains changing over time, and two weight vectors with L =16 (b) and L = 8 (c)

What if we further reduce the number of sample points? Select L = 8 as the number of observation points (i.e., the 13th, 15th, 29th, 35th, 47th, 53rd, 67th, and 69th domains here). Thus, the flow vector $\boldsymbol{x}$ has $L \times (N-1) = 8 \times 80 = 640$ variables. The related weight vector $\boldsymbol{S}$, calculated with 200 data points spanning two periods, is shown in Figure 8(c), which is almost the same as Figure 8(b). Next, with the observed data from only these eight sample points, we calculate the weight vector $\boldsymbol{S}$ using M data points within a moving time window MT from one time period to the next. Figure 9 shows the weight vector $\boldsymbol{S}$ evolving with T = 2.1s and the time window MT (= 200 × 2.1s = 420s) sliding ahead every 42s. With a few observation points visibility into time-varying network congestion appears indistinguishable during non-transient periods; however, we find that the effect of transient periods is very helpful for capturing the network-wide pattern shifting over time.

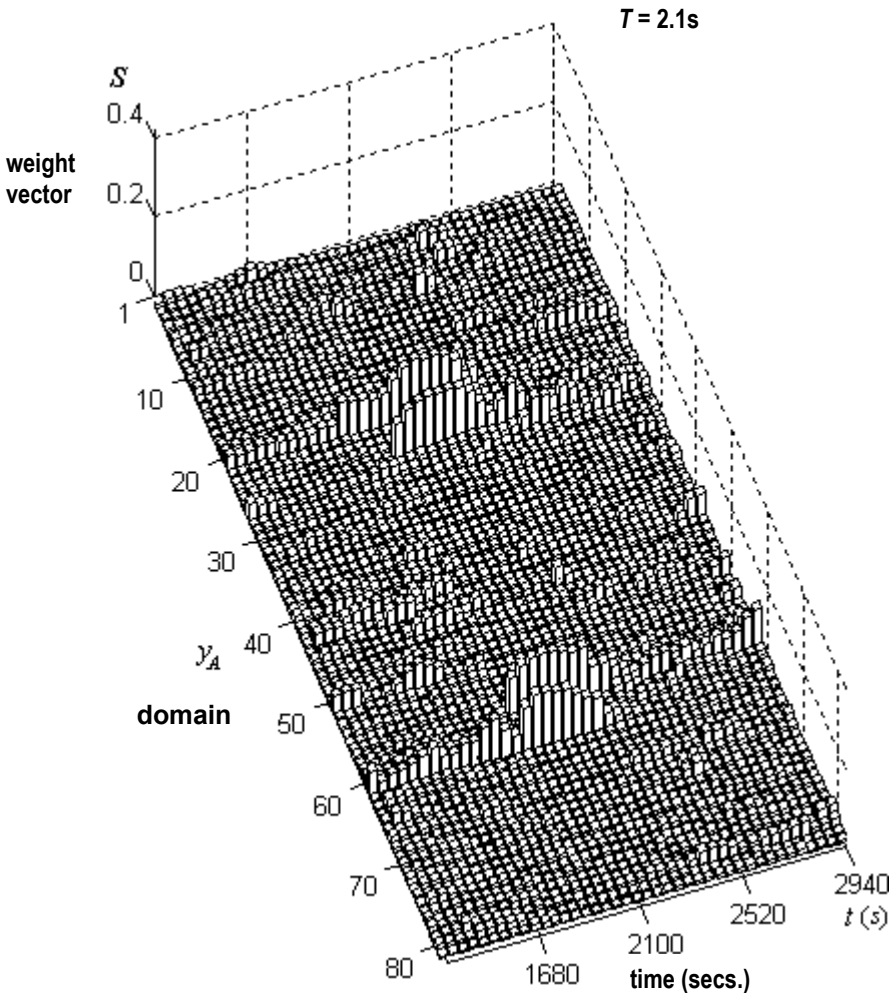

Figure 9: The spatial-temporal pattern observed with T =2.1s at eight observation posts

Can the proposed method succeed with still further reduction in the number of sample points? We finally select $L = 4$ as the number of observation points (i.e., the 31st, 33rd, 49th, and 51st domains here). The flow vector $\boldsymbol{x}$ has $L \times (N-1) = 4 \times 80 = 320$ variables. The weight vector $\boldsymbol{S}$, again calculated with 200 data points spanning two periods, is shown in Figure 10(a). Here, the performance of the method appears to degrade. While Figure 10(a) reveals the network-wide pattern to some extent, it also exhibits differences with Figure 8(b) and Figure 8(c). We attribute theses differences to local effects being amplified in the weight vector, but not appearing in the global pattern of Figure 8(b) and Figure 8(c). For example, S_{12} is very prominent in Figure 10(a), but not in Figure 8. This occurs because traffic from our four sampling domains to the 12th domain appears jammed because the routing algorithm in our model [11] forwards packets through the congested 21st domain. Despite degraded performance, the weight plot in Figure 10(a), though derived from only four sample points, is still helpful for inferring the network-wide pattern.

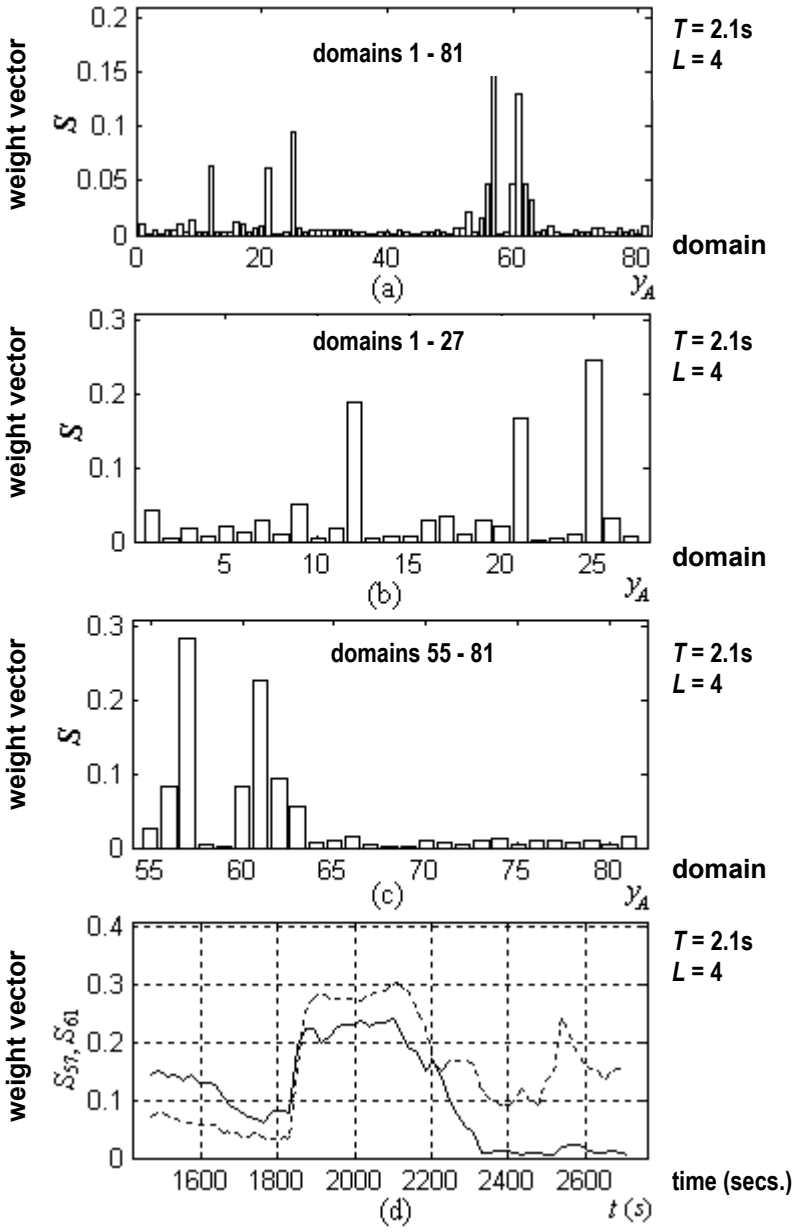

Figure 10: (a) a weight vector with L =4, (b) (c) two weight vectors for the first (1^{st} ~ 27^{th}) and third parts (55^{th} ~ 81^{st}), and (d) S_{61} (solid line) and S_{57} (dotted line) of the third part

Can we derive further insight by decomposing the network into parts with regard to the data analysis? We divide the network into three parts (i.e., 1^{st} ~ 27^{th}, 28^{th} ~ 54^{th}, and 55^{th} ~ 81^{st}), and analyze each separately. Since all hotspots exist in the first and third parts, Figure 10(b) and 10(c) show respectively their weight vectors, each of which is calculated with the flow vector of 4 × 27 = 108 variables. Notice that the weights of the domains are enhanced in these local maps. Figure 10(d) shows distinctly S_{61} (solid line) and S_{57} (dotted line) within the third part, which change with the moving time window MT (= 200 × 2.1s = 420s) ahead every 21s (recall Figure 9).

Our experiments suggest that we can gain network-wide knowledge of changing congestion patterns with substantially reduced data sets, but what effect does this reduced data have on computation requirements? Might we perform data analysis to support real-time monitoring? To produce Figure 10(c) requires just 0.06s for computing the correlation matrix, all eigenvalues and eigenvectors with MATLAB in a 1 GHz computer. Computing time requirements for our other computations were larger: 1.10s for Figure 10(a), 9.98s for Figure 8(c), and 82.92s for Figure 8(b).

4 Conclusion

Operating and engineering large-scale networks can benefit from development of promising tools to monitor network-wide traffic in space and time. In this paper, we proposed a cross-correlation method for spatial-temporal traffic analysis based on the eigenvector of the largest eigenvalue. To illustrate our proposed method and its potential promise, we reported results from some simulation experiments. Using a defined weight vector, we can identify the macroscopic pattern within the network at the critical timescale, and observe the more prominent weights of congested domains as increased correlation appears. In particular, we tried our method with various reductions in the number of observation points, and found that we could still capture the network-wide pattern shifting over time. We identified some degradation in the performance of our proposed method as the number of sample points passed below a threshold; however, we also found that we could compensate for this degradation somewhat by dividing the network into subareas and focusing on each smaller area separately.

While our proposed method appears promising, we have yet to apply the concepts to actual network measurements collected by members of the network research community. Another area for further investigation is comparison of our simulation results with the cross-correlations reported in the *Renater* study, which appear much stronger than those we find from our simulation model. For example, the largest eigenvalue reported in the *Renater* study is at least four times larger than the eigenvalues estimated from our simulations. This result might imply that actual traffic demand varies much more violently than the simulated "square wave" used in our experiments. If this implication holds, then our proposed method might be quite well suited for use in network operation and engineering. However, questions remain regarding how our ideas can be incorporated in practice into an operational network. For example, data recorded possibly by NetFlow [18] might need to be transmitted frequently to a collection server, and typically using unreliable UDP. Could such data collection induce measurement artifact into the cross-correlation eigenvalue depiction of the network?

Reference

[1] V. Paxson and S. Floyd, Wide-area traffic: The failure of Poisson modeling, in: Proc. ACM SIGCOMM '94, pp. 257-268, 1994.

[2] M. Crovella and A. Bestavros, Self-similarity in world wide web traffic: Evidence and possible causes, in: Proceedings of the 1996 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, May 1996.

[3] A. Feldmann, A. C. Gilbert, P. Huang, and W. Willinger, Dynamics of IP traffic: A study of the role of variability and the impact of control, in: Proc. ACM SIGCOMM '99, pp. 301-313, 1999.

[4] A. Feldmann, A. C. Gilbert, W. Willinger and T. G. Kurtz, The changing nature of network traffic: Scaling phenomena, ACM SIGCOMM Computer Communication Review 28 (2) (1998) 5-29.

[5] M. Crovella and E. Kolaczyk, Graph Wavelets for Spatial Traffic Analysis, in: Proceedings of IEEE Infocom 2003, San Francisco, CA, USA, April 2003.

[6] R. Mahajan, S. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker, Controlling high bandwidth aggregates in the network, Technical report, AT&T Center for Internet Research at ICSI, July 2001.

[7] P. Barford and D. Plonka, Characteristics of network traffic flow anomalies, in: Proceedings of ACM SIGCOMM Internet Measurement Workshop, San Francisco, CA, USA, November 2001.
[8] M. Barthelemy, B. Gondran, and E. Guichard, Large scale cross-correlations in Internet traffic, Physical Review E 66 (2002) 056110.
[9] V. Plerou, *et al.*, Random matrix approach to cross correlations in financial data, Physical Review E 65 (2002) 066126.
[10] K. Thompson, G. J. Miller, and R. Wilder, Wide-Area Internet Traffic Patterns and Characteristics, IEEE Network 11 (6) (1997) 10-23.
[11] J. Yuan and K. Mills, Simulating the Time-Scale Dynamics of Network Traffic Using Homogeneous Modeling, submitted, 2002.
[12] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2000.
[13] The MathWorks, Inc., Natick, MA, USA, MATLAB Ueser's Guide, 1998.
[14] H. J. Kim, Y. Lee, B. Kahng, and I. Kim, Weighted scale-free network in financial correlations, Journal of the Physical Society of Japan 71 (9) (2002) 2133-2136.
[15] K. I. Goh, B. Kahng, and D. Kim, Spectra and eigenvectors of scale-free networks, Physical Review E 64 (2001) 051903.
[16] J. Yuan, K. Mills, Exploring Collective Dynamics in Communication Networks, Journal of Research of the National Institute of Standards and Technology 107 (2) (2002)179-191.
[17] M. Grossglauser and D. Tse, A time-scale decomposition approach to measurement-based admission control, In: Proceedings of IEEE Infocom '99, pp. 1539-1547, New York, NY, March 1999.
[18] Cisco NetFlow. http://www.cisco.com/warp/public/732/netflow/index.html.